Attorney Docket No.: D02316-04 PATENT

IN THE UNITED STATES PATENT & TRADEMARK OFFICE

Inventor: Eric J. Sprunk	)	
U.S. Serial No.: 09/827,630	)	
Filed: April 6, 2001	)	Art Unit: 2135
	)	Examiner: Ponnoreay Pich

Title: AUTHORIZATION USING CIPHERTEXT TOKENS

DECLARATION UNDER 37 C.F.R. § 1.131

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir,

I, Eric J. Sprunk, hereby declare as follows:

1. I am the named and true inventor in the above referenced patent application and that I am the sole inventor of the subject matter disclosed and claimed in the above referenced patent application.

2. I submitted a description of my invention, now claimed in claims 1-7 and 11-14 of the above application, to the law department of General Instrument Corporation in an "Invention Record Form." I signed the Invention Record Form on October 5, 1999 and the signatures on the Invention Record Form are my own. A copy of the Invention Record Form is provided with this declaration as Attachment A. General Instrument Corporation Invention Record Form No. D02316CIP4.

3. I conceived the invention recited in claims 1-7 and 11-14 of the above application prior to June 2, 1998. The conception of the invention prior to this date is attested to in paragraph III(9) of the aforementioned General Instrument Corporation Invention Record Form No. D02316CIP4, and evidenced by the June 2, 1998 General Instrument Memorandum entitled "Application Security for TCI". This memorandum was referenced in and physically attached to General Instrument Corporation Invention Record Form No. D02316CIP4 when the form was witnessed by Alexander Medvinsky, a General Instrument Corporation employee, on November 5, 1999. See Attachment A.

4. I constructively reduced my invention to practice prior to June 2, 1998, and this reduction was memorialized in the aforementioned "Application Security for TCI" memorandum. This memorandum was provided to fellow General Instrument employees Paul Moroney, Gary Albeck, B. Meandija, Petr Peterka, Xin Qui, Stuart Moskovics, Steven Anderson, K. Miller, J. Fellows, Annie Chen, Lawrence Tang, Mark DePietro, Douglas Makofka, Reem Safadi, and Lawrence Vince (as evidenced by the distribution list on the face of the memorandum).

5. Upon information and belief, the date of receipt of General Instrument Corporation Invention Record Form No. D02316CIP4 by the General Instrument Corporation law department was October 8, 1999, as evidenced by the "General Instrument Corporation Intellectual Property" date stamp on the first page of Attachment A.

6. I hereby declare that all statements made herein based upon knowledge are true, and that all statements made based on upon information and belief are believed to be true,. These statements were made with the knowledge that willful false statements and

the like so made are punishable by fine or imprisonment, or both, under § 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Dated: 27 Jan 2006 By: _____
Eric J. Sprunk

APPENDIX A

General Instrument Corporation Invention Record Form No. D02316CIP4
Inventor: Eric J. Sprunk

U.S. Serial No.: 09/827,630

General Instrument Corporation®
Intellectual Property Department
For Internal Use Only

Invention Record Form
GI Docket No. 12316 CIP4

18936-003160

I. Administrative Information

1. Short Descriptive Title of the Invention: ~~Objects 1 & 2 Secure Search~~ Authorization Using Ciphertext Tokens

2. Identify all persons who contributed to this invention, including persons from other divisions and/or outside companies:

	Inventor 1	Inventor 2
Full Legal Name	Eric Sprunk	
Home Address	6921 Cayenne Lane	
City, State, Zip	Carlsbad, CA 92009	
Citizenship	US	
Division/Co. Location	AT (SD)	
Office Phone No.	858-404-2426	
Mgr.'s Name & Phone No.	Maroney 858-404-2445	
Signature of Inventor		
Date	10/5/99	

	Inventor 3	Inventor 4
Full Legal Name		
Home Address		
City, State, Zip		
Citizenship		
Division/Co. Location		
Office Phone No.		
Mgr.'s Name & Phone No.		
Signature of Inventor		
Date		

3. ☐ Check box if there are additional inventors listed on separate sheets. Additional information concerning inventors, if any

U.S. Serial No.: 09/827,630

Invention Record Form

II. Background Information

1 Do you believe this invention was developed while working under or in the performance of experimental, developmental or research work called for by a government contract or with the understanding that a government contract would be awarded? ☒ No ☐ Yes If yes, please explain:

2. Has your invention been disclosed to anyone outside General Instrument in a speech, exhibit, presentation, product, product manual, report, lecture, trade show, technical article, publication or otherwise? ☐ No ☒ Yes If yes, please explain:
ATT/TCI - June 98 - UNDER NDA

3 Is this invention related to any previous GI invention disclosures of which you are aware (made by you or someone else)? ☐ No ☒ Yes If yes, please explain:
D2303 D2315 D2310 } Conditional inventions
D2311 D2317 D2312 }

4 Name of product(s) and/or project(s) for which this invention was developed:
DCT 5000

5 Planned or actual use of invention:
Various stages - from 4Q99 through 2000

6. What economic benefits do you think GI can derive from this invention?
Potential that all CA over network (computers, RD, etc) may implement this CA/security model. Copy protection ring would be [illegible]

7. When do you expect a product incorporating this invention to be sold, offered for sale or shown to someone outside of GI? (If a product or prototype has already been sold, offered for sale or shown, please identify the earliest date this happened.)
4Q 99 for some orders.

8 Has a working model of the invention been built and tested (or appropriate software been written)?
☐ No ☑ Yes If yes, who has witnessed a demonstration, and when?
Some requests for 5000 early Sept. 99

[signature] 10/5/99
Signature of Submitter(s) Date

[signature] 10/5/99
Read and understood by (Witness Signature(s)) Date

Invention Record Form

9. List below any patents, publications, articles, texts, products, etc. which describe technology similar to your invention including reference material which may be useful in understanding the background technology of your invention (Use a separate sheet if necessary and attach a copy of each item. Please include copies of all bibliographical information.) (Use a separate sheet if necessary)

Only JAVA type applications may be relevant.
Nothing specific known

Signature of Submitter(s)

10/5/99
Date

Read and understood by [Witness Signature(s)]

10/13/99
Date

U.S. Serial No.: 09/827,630

Invention Record Form

III. Description of the Invention

1 Please provide a very brief (i.e., one short sentence) summary of your invention.

System for applying CA to [illegible] software objects & resources
System applying CA to object/resource instead of channel

2. Briefly describe the field of technology to which your invention relates.

Conditional access & Security

3. Briefly describe the problems, issues or needs which led to the invention

Control of individual objects & resources rather than one big [illegible]
Control with respect to time [illegible]
Control with respect to different platforms/systems

4. How have others addressed these problems, issues or needs?

[illegible]

5. Describe those particular features or functions of your invention which you think may be novel or technical advancements over the technology you listed in section II.9.

Attaching/Associating CA to particular objects/resources
for set time
for different platforms/systems

6 Best Mode: Describe any and all preferences you personally have regarding how to best implement, build, produce or use your invention (e.g., preferred parts, materials, techniques, etc. which you feel are best in practicing your invention). Each submitter's opinion is important here, even if there is disagreement. Please list anything you think will make the invention better in any way

As described in memo

7. Briefly describe any alternative uses, variations or modifications of your invention which you contemplate.

described in memo

8. Please provide any additional information you think should be known by the attorney reviewing this form.

None

[signature]
Signature of Submitter(s)

10/5/99
Date

[signature]
Read and understood by [Witness Signature(s)]

10/5/99
Date

GI CONFIDENTIAL & PROPRIETARY

Rev 02/98

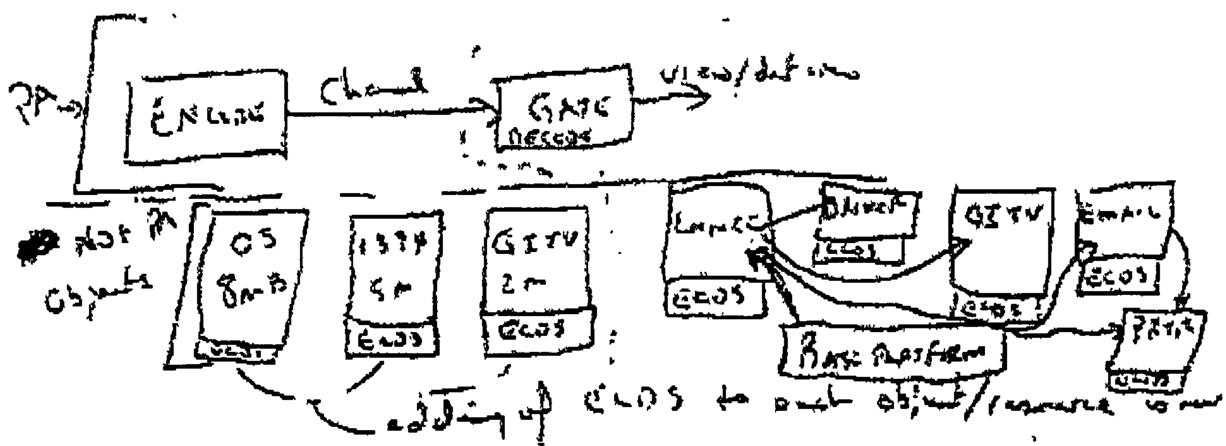
U.S. Serial No.: 09/827,630

Invention Record Form

9 Please provide a detailed description of your invention. Your description should ideally provide as many details of your invention as possible in order to achieve optimal patent protection. An ideal disclosure should describe the construction and operation of the invention including drawings (flow charts, schematics, block diagrams, mechanical drawings, photographs, etc.) and any relevant engineering laboratory notebook pages, reports, program listings, etc. If you have already prepared reports or other descriptive information, there is no need to rewrite it. Simply attach it and reference it in your invention disclosure data sheet (for example, "see attached 9 page engineering progress report addressed to John Doe dated 1 Jan., 1992 for description of amplifier circuit")

See attached 6/2/98 Memo and more specifically, security level 8 on pg. 9.

{Focus on Record of 6/2/98 & 6/11/98 → from Eric Sprunk to P. Moroney et al & Sid Gregory, respectively. Both Subject lines Application Security (Crypto) for TCI



(Origin)
Resource - anything used by an object to operate as intended — may be another object or a physical device (product).

Signature of Submitter(s) — 10/5/99 Date

Read and understood by (Witness Signature(s)) — 10/5/99 Date

 Rev 02/98

U.S. Serial No.: 09/827,630

10:51AM

Memorandum General Instrument

Date: June 2 1998

Subject: Application Security for TCI

From: Eric Sprunk

To: P Moroney, G Albeck, B Mesmdija, P. Peterka

CC: X Qiu, S Moskovics, S. Anderson, K. Miller, J. Fellows, A. Chen, L. Tang, M. DePietro, D Makofka, R. Saindi, L. Vinos

1. Introduction

This memo provides background to facilitate the definition of TCI requirements (and GI design decisions) associated with applications security on the DCT5000 (hereafter the "5000") product

1.1 Acronyms & Abbreviations

5000 TCI's DCT-5000 advanced setop

A&A Authorization and Authentication, in that order

ACP Access Control Processor

App Application

BIOS Built-in Operating System

CA Conditional Access or Certificate Authority

EMM Entitlement Management Message

ECDS Entitlement Control Data Structure

ECM Entitlement Control Message

ET Execution Token

IVV Independent Validation and Verification

JVM Java Virtual Machine

OS Operating System

2. Security Environments

2.1 The Video Service Security Model

A video service is a continuous stream of data consisting of individual program segments. Different security techniques apply control to this situation:

- Encryption is used through the possession of a valid key.
- The encrypted stream is routed through an Access Control Processor (ACP) security device.
- The ACP only decrypts the service if:
  - it has a valid key; and
  - ECM information passes certain data checks (or gates), e.g. possessing a specific tier
- The encryption key used is changed regularly to facilitate this, e.g. hourly or monthly

The placement of the ACP in series with the data path is crucial, as this makes its "gatekeeper" functionality possible. Were the ACP not in series with the data stream, such as with a typical DVB smart card system, then the security control effected by the ACP would need to reach outside it to another setop component. This creates security risks avoided by merging MPEG security processing with Conditional Access.

It remains possible to inject clear data downstream of the ACP, and downstream non-ACP circuitry will accept and process such data normally. With the exception of minor security mechanisms like

Macrovision copy protection, the control of security is limited to within the ACP. Components outside the ACP cannot be depended upon to enact security functions

Such a configuration seeks to control the entire data stream as its first objective, and a given piece of data from that data stream (e.g. a program) as a second objective. The ability to do this comes from the continuous series of checks performed on each program, which each presents the opportunity to change a key or tier data associated with the program. If keys or tiers are changed for the headend ACP that encrypts the service, then a setop ACP has no choice but to do the same to decrypt

But a specific program is really "gated" only once. If ever the ACP makes the decision to decrypt that program, then all ability to control the program is thereafter lost with this creation of clear data. If that clear program is stored (on a sufficiently large media), then it will be available forever Aside from the possible unavailability of large storage media, encryption control is binary in nature and impossible to recover once lost. Consequentially, post-ACP injection of clear video data will be successful, whether that data was recorded from earlier decryption or from some never-encrypted source of MPEG data.

This existing security model for video has limitations when applied to applications. The discussion below highlights these differences one at a time, in the context of known or implied TCI requirements.

2.2 Application Security

An application (or "App" hereafter) has characteristics in common with a video program which allow video types of security control to work acceptably for purposes that follow the video model. However, the difference between an App and a video program gives rise to new problems in need of new security solutions There are a number of these, with only partly satisfactory solutions available for some problems

An obvious example is how Apps differ from video in the size of their data. Video data streaming at even 1Mbps for a one hour program comprises 450 MB of information, storage of which tends to be impractical at present. This storage problem presently serves as a barrier to replaying video data But, an App is comparatively tiny at less than one megabyte, and storage is clearly feasible. The replay of old Apps or the injection of new Apps will be easier than for video data, and may therefore be a more significant problem.

This problem alone illustrates how App security techniques must be extended out beyond the ACP. It is a given that the ACP cannot undertake all the functions of the entire 5000 in a single chip, though this may be possible some day for a low end setop. Until such a "one chip setop" exists, new security techniques will be needed to deal with Apps. It will not be sufficient for an ACP to serve only as a data stream gatekeeper, as this will not address the newly significant problem of data replay and control outside the ACP.

Some extensions of ACP security to outside the ACP are easy to identify, and have identifiable security benefits and limitations in addressing some requirements. For others this is not so straightforward, and careful consideration of requirements, the practicality of available security solutions, security benefits, and limitations is needed. Some App security problems are very difficult to counter without difficult and significant development efforts A discussion of value and possible diminishing return is paramount.

This memo elucidates multiple possible levels of App security on the 5000. The models are described starting at the same level as video security, then through several increasing levels of protection. The models are listed sequentially, and are taken directly from or heavily implied in the TCI DCT5000 specification. In some cases, the requirement listed is derived from GI interpretation of TCI's high

level security goals for the 5000. The definition of each requirement is in italics, followed by discussion in normal typeface.

3. The Levels of Application Security

The single high level requirement to "secure Applications" can readily be resolved into a number of specific sub-levels to consider one at a time. In general, only the first of these levels can be attained without extending security outside the ACP. Further, an Entitlement Control Data Structure[1] (ECDS) appended to each App is needed for any Authorization functions to occur, and includes a digital signature. The Operating System (OS) must make security checks at various times using this data structure, and use the result of the check as a hard decision on certain OS functions. Several possible security levels are possible, with the assumed level of trust of the OS itself affecting them all. The size of the ECDS must be considered, as well as system decisions such as repetitive download of a single App or its ECDS for security purposes. It is also necessary to distinguish the initial launch of an App from its continuous execution afterwards. These possibilities are all discussed below. Each security level includes the functions of levels below, e.g. Level 4 App Security includes all techniques and protections described for Levels 1, 2, and 3.

3.1 Level 1: Encrypted Application Download

This security level is defined as controlling the entry of an App into a settop via cable data pathways. Download is here defined as being the movement of MPEG data over the FDC or OOB, and does not include the injection of data directly into App memory by another path.

This level is the easiest to attain, since a stream of data comprising an App can be treated identically to a video stream. The App stream may be inband or out of band, but it can be simply placed on an MPEG PID in encrypted form, with associated ECMs. The single EMM stream that conveys all entitlements to individual settops is used to convey encrypted App entitlements as well.

The encrypted App passes through the ACP[2] in the normal manner, and decryption occurs only if authorized[3]. Clear, Fixed Key, and Full Encryption modes would be available.

3.2 Level 2: Download Authentication

Authentic Apps are here defined as Apps that are approved by the network operator using a digital signature.

The network operator would authenticate an App by processing it in the headend or elsewhere, and by appending such authentication to the App via the ECDS. Such network operator authentication includes the entitlements or authorizations needed to use that App. It is absolutely crucial that only the network operator be capable of authenticating an App.

---

[1] The Entitlement Control Data Structure is analogous to an ECM for a video service, and conveys the entitlements needed by an ACP to authorize that specific App.

[2] Note that routing the OOB MPEG data through the ACP has both ACP and setup design implications. The ACP must receive MPEG transport both from the inband and OOB sources, which requires that it have two MPEG inputs, or that MPEG data be multiplexed outside the ACP.

[3] Encrypted download has a form of authentication, where only the possessor of the encryption key can mark an App as authentic, or confirm authenticity. This type of authentication is implicit, and may be sufficient for a form of basic protection.

A digital signature is the obvious way of achieving this, with a network operator control computer holding the key that creates this signature[4]. Either symmetric (e.g. DES based) or asymmetric (e.g. RSA or DSA or ECC based) signatures would work, but asymmetric signatures offer the best solution. The choice of asymmetric signature type must be made based on speed, signature size, licensing, and other considerations not discussed in this section.

Defining an Authentication mechanism does not address the circumstances under which the Authentication is confirmed. The App can be checked for validity at different times that define different security levels. These are enumerated in sections below. *Application Security Level 2 only assumes basic authentication, such as right after download decryption and before the App is loaded into storage.*

Since App signature verification is almost inescapably linked to Authorization, it must occur within the ACP. Signature verification in the ACP also minimizes the burdensome impact of hash and signature functions on the same CPU that performs video, administrative, or GUI functions. Secure confirmation and reportback of signature verification failures are important, and there is an implied requirement for Return Path capability. Setops without RF or phone modem reportback capability cannot be monitored for App security behavior, and represent a higher level of risk[5].

The consideration of Authentication brings us to the most important issue in App security. Current approaches[6] to Authentication have the OS itself performing verification and enforcement in the event of failure. This is because the security benefit of Authentication is inherently dependent upon the trust level of the OS. One might think that an ACP can block execution of an App by the OS, but this is untrue so long as OS design and operation is itself beyond ACP secure control. If the OS circumvents an Authentication check, then there is nothing whatsoever that the ACP can do about it. In fact, the ACP would not even be aware of such an event. The trust level of the OS will be a recurring theme in this discussion, and will be returned to again in a subsequent section.

3.3 Level 3: Authenticated Launch

Authenticated Launch is defined as the initiation of execution of an App by the OS only if it is network operator approved.

This level is where the Authentication defined in the previous section is verified by the OS, using the ACP, at App launch time. Authentication of launch requires the following steps:

1. The OS loads 100% of App data into the ACP. The signature is not initially included. The OS must not execute the App until the ACP has confirmed that the signature is valid.
2. The ACP forms the Message Authentication Code (MAC) using a hash function.
3. The OS separately loads the App signature into the ACP.

---

[4] At this security level, Authentication functions are not present for an Application. An App digital signature would thus only authenticate the App itself. At a higher security level described later, both the App and its authentication requirements would be included in the digital signature.

[5] Should Applications be allowed in devices such as this? Perhaps the ability to prohibit them in the event of future problems is needed, or at least allow problem setops (with a hacked OS) to be identified if necessary.

[6] Microsoft Authenticode is a digital signature scheme with Microsoft taking on the role of the network operator. MS signs the applications using Authenticode, and MS OS products confirm that signature as a pre-condition for using the MS Crypto API, rather than as a pre-condition for launching a program. MS Authenticode does not perform any Authorization functions.

4. The ACP checks the digital signature, responding with VERIFIED or FAILED
5. If VERIFIED, the OS initiates execution of the App.
6. If FAILED, the OS erases the App from executable memory

3.4 Level 4: Authorized Launch

Authorized Launch is defined as the initiation of execution of an App by the OS only if appropriate encryption keys and entitlements are possessed by that specific setop ACP

Authorization of launch can be achieved by at least two approaches with different levels of practicality. An App is always stored in authenticated form, but it could be stored either in encrypted form, or in clear form. (Note that, if stored in encrypted form, the need for encryption during download may be obviated.) If Apps are stored encrypted, then they must be decrypted to allow execution. This requires that:

1. the OS load all App data into the ACP
2. the ACP decrypt all App data.
3. the ACP hand all decrypted App data back to the OS.

The first step of this 3 step process occurs during Authentication, but the second and third steps do not and represent additional work. Authentication takes care of perhaps 50% of the work of decryption.

The security benefit of encrypted storage is dependent upon the existence of Authentication:

- Lacking Authentication, encrypted storage prevents injection of unapproved Apps into the setop, since the encryption key is not (normally) available for illicit use.
- With Authentication present, encrypted storage has value only if the OS is untrustworthy.

But this is a contradiction. If the OS is untrusted, then Authentication has no value, so encrypted storage actually adds nothing to security. This conclusion is typically illustrative of OS importance in evaluating App security. *Almost all possible App security fails if the OS is not trustworthy.* Encrypted storage is not recommended, assuming (inescapable) Authentication is present.

Assuming no encrypted storage, Authorization of launch requires the following steps:

1. The OS loads the ECDS into the ACP.
2. The ACP checks whether it possesses the entitlements and keys necessary to run that App
3. The ACP advises the OS of AUTHORIZED or NOT AUTHORIZED status.
4. The OS loads and begins execution of the App if AUTHORIZED.
5. The OS erases the App and does not execute if NOT AUTHORIZED.

Recall that this security level presumes the protections present in lower security levels, including Authentication. Both Authentication and Authorization checks must occur before launch, but in which order? In general, Authentication will require much more time to perform than Authorization, so determination of NOT AUTHORIZED status is faster if authorization is done first.

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☑ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.